IN THE CLAIMS:

Please cancel claims 1-35. Please add claims 36-70 as indicated below. A complete listing of claims follows.

1-35. (Canceled)

36. (New) A microprocessor, comprising:

an instruction cache configured to fetch instructions for execution; and

a return prediction unit coupled to said instruction cache, wherein said return prediction unit comprises:

a return storage including a first entry configured to store a count value and a return address of a previously detected call instruction; and

a controller coupled to the return storage and configured to compare the return address stored in the first entry to a new return address of a subsequently detected call instruction, and in response to determining that the stored return address is the same as the new return address, to modify the count value to indicate that the stored return address corresponds to more than one call instruction;

wherein if the count value indicates that the stored return address corresponds to more than one call instruction, the controller is configured to provide a predicted return address to the instruction cache for fetching by providing the stored return address and correspondingly modifying the count value.

37. (New) The microprocessor as recited in claim 36, wherein in response to determining that the return address stored in the first entry is the same as the new return address, the controller is configured to modify the count value without allocating a new entry within the return storage.

38. (New) The microprocessor as recited in claim 36, wherein the controller is configured to modify the count value to indicate that the stored return address corresponds to more than one call instruction by increasing the count value.

39. (New) The microprocessor as recited in claim 36, wherein if the count value indicates that the stored return address corresponds to more than one call instruction, the controller is configured to correspondingly modify the count value by decreasing the count value.

40. (New) The microprocessor as recited in claim 36, wherein if the new return address is not the same as the stored return address, the controller is configured to allocate a new entry within the return storage for the new return address.

41. (New) The microprocessor as recited in claim 40, wherein the new entry is configured to store a count value and wherein the controller is configured to initialize the count value of the new entry to a minimum count value.

42. (New) The microprocessor as recited in claim 36, wherein if the new return address is the same as the stored return address and the count value equals a maximum count value, the controller is configured to allocate a new entry within the return storage for the new return address.

43. (New) The microprocessor as recited in claim 36, wherein the return storage is implemented as a stack structure, and wherein the first entry is identified by a top of stack pointer.

44. (New) The microprocessor as recited in claim 43, wherein if the new return address is not the same as the stored return address, the controller is configured to allocate a new entry for the new return address and to modify the top of stack pointer to identify the new entry.

45. (New) The microprocessor as recited in claim 43, wherein in response to a branch prediction being made, the controller is configured to save a copy of a current value of the top of stack pointer.

46. (New) The microprocessor as recited in claim 45, wherein the controller is further configured to save a copy of the count value associated with the first entry if the first entry is identified by the top of stack pointer when the branch prediction is made.

47. (New) The microprocessor as recited in claim 36, wherein each entry in the return storage is configured to store a respective count value.

48. (New) The microprocessor as recited in claim 36, wherein fewer than all entries in the return storage are configured to store a respective count value.

49. (New) The microprocessor as recited in claim 36, wherein the controller is further configured to provide the predicted return address to the instruction cache for fetching in response to a return instruction being detected.

50. (New) The microprocessor as recited in claim 49, wherein if the count value associated with the stored return address is equal to a minimum value after being correspondingly modified, the controller is further configured to remove the first entry from the return storage.

51. (New) The microprocessor as recited in claim 36, wherein if the count value indicates that the stored return address corresponds to a single call operation, the controller is configured to provide the return address by providing the stored return address and removing the first entry from the return storage.

52. (New) A method, comprising:

storing a count value and a return address of a previously detected call instruction in a first entry of a return storage;

comparing the return address stored in the first entry to a new return address of a subsequently detected return instruction;

in response to determining that the stored return address is the same as the new return address, modifying the count value to indicate that the stored return address corresponds to more than one call instruction; and

if the count value indicates that the stored return address corresponds to more than one call instruction, providing a predicted return address for fetching by providing the stored return address and correspondingly modifying the count value.

53. (New) The method as recited in claim 52, wherein modifying the count value to indicate that the stored return address corresponds to more than one call instruction occurs without allocating a new entry within the return storage.

54. (New) The method as recited in claim 52, wherein modifying the count value to indicate that the stored return address corresponds to more than one call instruction includes increasing the count value.

55. (New) The method as recited in claim 52, wherein correspondingly modifying the count value if the count value indicates that the stored return address corresponds to more than one call instruction includes decreasing the count value.

56. (New) The method as recited in claim 52, further comprising allocating a new entry within the return storage for the new return address in response to determining that the stored return address is the same as the new return address

57. (New) The method as recited in claim 56, wherein the new entry is configured to store a count value and wherein the method further comprises initializing the count value of the new entry to a minimum count value.

58. (New) The method as recited in claim 52, further comprising allocate a new entry within the return storage for the new return address in response to determining that the new return address is the same as the stored return address and the count value equals a maximum count value.

59. (New) The method as recited in claim 52, wherein the return storage is implemented as a stack structure, and wherein the first entry is identified by a top of stack pointer.

60. (New) The method as recited in claim 59, further comprising allocating a new entry for the new return address and modifying the top of stack pointer to identify the new entry in response to determining that the new return address is not the same as the stored return address.

61. (New) The method as recited in claim 59, further comprising saving a copy of a current value of the top of stack pointer in response to a branch prediction being made.

62. (New) The method as recited in claim 61, further comprising saving a copy of the count value associated with the first entry in response to determining that the first entry is identified by the top of stack pointer when the branch prediction is made.

63. (New) The method as recited in claim 52, wherein providing the predicted return address to the instruction cache for fetching occurs in response to a return instruction being detected.

64. (New) The method as recited in claim 63, further comprising removing the first entry from the return storage in response to determining that the count value associated with the stored return address is equal to a minimum value after being correspondingly modified.

65. (New) A system, comprising:

a system memory; and

a microprocessor coupled to the system memory, wherein the microprocessor includes:

an instruction cache configured to fetch instructions for execution; and

a return prediction unit coupled to said instruction cache, wherein said return prediction unit comprises:

a return storage including a first entry configured to store a count value and a return address of a previously detected call instruction; and

a controller coupled to the return storage and configured to compare the return address stored in the first entry to a new return address of a subsequently detected call instruction, and in response to determining that the stored return address is the same as the new return address, to modify the count value to indicate that the stored return address corresponds to more than one call instruction;

wherein if the count value indicates that the stored return address corresponds to more than one call instruction, the controller is configured to provide a predicted return address to the instruction cache for fetching by providing the stored return address and correspondingly modifying the count value.

66. (New) The system as recited in claim 65, wherein in response to determining that the return address stored in the first entry is the same as the new return address, the controller is configured to modify the count value without allocating a new entry within the return storage.

67. (New) The system as recited in claim 65, wherein the controller is configured to modify the count value to indicate that the stored return address corresponds to more than one call instruction by increasing the count value.

68. (New) The system as recited in claim 65, wherein if the count value indicates that the stored return address corresponds to more than one call instruction, the controller is configured to correspondingly modify the count value by decreasing the count value.

69. (New) The system as recited in claim 65, wherein if the new return address is not the same as the stored return address, the controller is configured to allocate a new entry within the return storage for the new return address.

70. (New) The system as recited in claim 65, wherein the return storage is implemented as a stack structure, wherein the first entry is identified by a top of stack pointer, and wherein if the new return address is not the same as the stored return address, the controller is configured to allocate a new entry for the new return address and to modify the top of stack pointer to identify the new entry.